

Webové služby **DRAŽBY**

webové služby pro služby OctopusPro

OctopusPro s.r.o.

Verze: 1.0.2

Datum poslední aktualizace: 24.10. 2017

Obsah:

1	Úvod.....	3
1.1	Základní popis.....	3
2	Implementace webových služeb OctopusPro	3
2.1	Základní třídy požadavku a odpovědi (RequestBase a ResponseBase).....	3
2.2	Příklad použití WS.....	4
3	webové služby pro DRAŽBY	9
3.1	Seznam použitých webových služeb.....	9
3.2	ws_drazba_2c.....	9
3.2.1	Ukázka volání funkce ws_drazba_2c	9
3.3	ws_octop_get_AvailableUsers	11
3.4	ws_octop_get_LOV_Region.....	11
3.5	ws_octop_get_LV_SubRegion	11
4	Datové typy - Data transfer objekty	12
4.1	tHitDrazbaDTO.....	12
4.2	tSearchProfileDrazbaRequestDTO.....	12
4.3	tSearchProfileCategoryRequestDTO.....	13
4.4	tSearchProfileRegionRequestDTO	13
4.5	tUserDTO	13
5	Číselníky.....	13
5.1	CC_Category	13
5.2	CC_Region.....	14
5.2.1	Abecední řazení	14
5.3	CC_SubRegion.....	14
6	Ukázka čistého xml dotazu na webovou službu ws_drazba_2c	16

1 ÚVOD

Úkolem webových služeb poskytovaných společností OctopusPro s.r.o. je zpřístupnění dat pro software třetích stran. Jejich využívání je podmíněno vytvořením aplikace na straně klienta, který bude poskytnutá data dále zpracovávat.

1.1 Základní popis

Webové služby jsou veřejně dostupné na následující adrese.

<http://publicws.octopuspro.cz/OctopusProService.svc?wsdl> – služby pro monitoring, dražby a párování

K úspěšnému přihlášení do webové služby je nutné zažádat o přístupové údaje přímo u společnosti OctopusPro s.r.o. Správné nastavení služeb je dále popsáno v příkladech, případně v kapitole Nastavení služeb.

2 IMPLEMENTACE WEBOVÝCH SLUŽEB OCTOPUSPRO

2.1 Základní třídy požadavku a odpovědi (RequestBase a ResponseBase)

Všechny webové služby mají jeden vstupní a jeden výstupní parametr.

Vstupní parametr tzv. „request“ parametr je odvozený od základní třídy RequestBase. Výstupní parametr tzv. „response“ je odvozený od třídy ResponseBase. Popis jednotlivých parametrů obou následuje:

RequestBase – základní třída pro vstupní parametr web.služby

Název elementu	Typ	Možné hodnoty	Poznámka
ClientTag	string	Neprázdný string	Identifikace klienta WS – ClientTag je přidělen společně s přístupovými údaji k WS
ClientUsername	string	Neprázdný string	Uživatelské jméno klienta – přístupové údaje k WS
ClientPassword	string	Neprázdný string	Uživatelské heslo klienta – přístupové údaje k WS
RequestId	string	Jednoznačná identifikace daného požadavku	Unikatni GUID requestu, který ho identifikuje, např. 113ADD42-171D-45ED-A275-E34AD369EFBB
LoadOptions	string[]	Případná hodnota se nastavuje dle konkrétní volané služby	Nepovinné pole parametrů udávající doplňující požadavky na vrácená data. Např. načtení číselníku k vráceným datům atd.
Action	String	Případná hodnota se nastavuje dle konkrétní volané služby	Nepovinné pole, pomocí něhož je možné vyvolat speciální akce definované u jednotlivých služeb

ResponseBase – základní třída pro výstupní parametr web.služby

Název elementu	Typ	Možné hodnoty	Poznámka
Acknowledge	AcknowledgeType	Success, Failed	Udává výsledek volání webové služby. V případě chyby vrací Failed
Message	String	Prázdný string v případě úspěchu	V případě chyby (Acknowledge) je zde možné nalézt detailnější popis, proč se volání nezdařilo

ComparationId	string	GUID příslušného requestu	Unikátní GUID requestu, pro který byl vytvořen tento response
RowsAffected	int	-	Počet navrácených záznamů, pokud to vzhledem k charakteru WS dává smysl
Version	string	-	Verze WS
Build	string	-	Build WS
ReservationId	string	-	Vyhrazeno pro dlouhotrvající requesty, zatím nepoužito
ReservationExpires	Datetime	-	Vyhrazeno pro dlouhotrvající requesty, zatím nepoužito

2.2 Příklad použití WS

Všechny příklady jsou uvedeny v jazyce C# (.NET Framework). Princip je ale stejný ve všech programovacích jazycích.

Nejdříve je potřeba přidat WS do Service References v daném projektu. V následujících ukázka reference odpovídá "OctopusServiceClient"

Ukázka nastavení připojení k WS z web.config

```
<system.serviceModel>
  <bindings>
    <basicHttpBinding>
      <binding name="basicHttpEndpoint" closeTimeout="00:10:00" openTimeout="00:10:00"
        sendTimeout="00:10:00" maxBufferPoolSize="2147483647" maxReceivedMessageSize="2147483647" />
    </basicHttpBinding>
  </bindings>
  <client>
    <endpoint address="http://publicwstest.octopuspro.cz/OctopusProService.svc"
      binding="basicHttpBinding" bindingConfiguration="basicHttpEndpoint"
      contract="OctopusService.IOctopusService" name="basicHttpEndpoint" />
  </client>
</system.serviceModel>
```

Pro zjednodušení tvorby jednotlivých requestů je vhodné vytvořit mechanismus pro automatické naplnění všech potřebných parametrů z třídy RequestBase (přístupové údaje). K tomu může sloužit následující statická třída:

```
/// <summary>
/// Request helper pro přístupové údaje uložené ve web.config
/// </summary>
public static class RequestHelper
{
  /// <summary>
  /// Client Tag.
  /// </summary>
  public static string ClientTag { get; private set; }
  /// <summary>
  /// Uživatelské jméno
  /// </summary>
  public static string ClientUsername { get; private set; }
  /// <summary>
  /// Uživatelské heslo
  /// </summary>
}
```

```
public static string ClientPassword { get; private set; }

/// <summary>
/// .ctor - načte přístupové údaje z web.config
/// </summary>
static RequestHelper()
{
    ClientTag = ConfigurationManager.AppSettings.Get("ClientTag");
    ClientUsername = ConfigurationManager.AppSettings.Get("ClientUsername");
    ClientPassword = ConfigurationManager.AppSettings.Get("ClientPassword");
}

/// <summary>
/// Metoda pro přípravu requestu (doplnění přístupových údajů a vytvoření unikátního GUID)
/// </summary>
/// <typeparam name="T"></typeparam>
/// <param name="request"></param>
/// <returns></returns>
public static T Prepare<T>(this T request) where T : RequestBase
{
    request.RequestId = RequestId;
    request.ClientTag = ClientTag;
    request.ClientUsername = ClientUsername;
    request.ClientPassword = ClientPassword;

    return request;
}

/// <summary>
/// Vrátí unikátní GUID
/// </summary>
public static string RequestId
{
    get { return Guid.NewGuid().ToString(); }
}
}
```

Dále je vhodné vytvořit třídu, která se bude starat o vlastní přístup k WS a bude nám vracet požadované výsledky. V dalším příkladu uvádíme ukázkou takové třídy pro přístup k webovým službám párování soukromé inzerce. Tato třída je nazvána jako **OctopusProServiceRepository**.

```
public class OctopusProServiceRepository
{
    /// <summary>
    /// Klient webové služby (Lazy load)
    /// </summary>
    protected OctopusServiceClient Client
    {
        get
        {
            // Zkontroluj jestli klient už nebyl inicializován
            if (HttpContext.Current.Session["OctopusServiceClient"] == null)
                HttpContext.Current.Session["OctopusServiceClient"] = new OctopusServiceClient();

            // vytvoř novou instanci, pokus se nepodařilo načíst ho ze session
            var client = HttpContext.Current.Session["OctopusServiceClient"] as OctopusServiceClient;
            if (client.State == CommunicationState.Faulted)
            {
                try { client.Abort(); }
                catch { /* no action */ }

                client = new OctopusServiceClient();
                HttpContext.Current.Session["OctopusServiceClient"] = client;
            }

            return client;
        }
    }
}
```

```
}

protected void Correlate(RequestBase request, ResponseBase response)
{
    if (request.RequestId != response.ComparationId)
        throw new ApplicationException("RequestId a ComparationId nesouhlasí.");
}

/// <summary>
/// Test přihlášení se (a připojení) k WS
/// </summary>
/// <param name="idUser"></param>
/// <returns></returns>
public ResponseBase TryLogin(int idUser)
{
    //vytvoření requestu, pro vyzkoušení přihlášení se k WS
    var request = new ws_tryLogin_Request().Prepare();

    //nastavení přístupových práv se kterými se chceme k WS připojit
    request.AccessType = SpecialAccessType.MonitoringSoukromaInzerce;
    //nastavení uživatele, který se k WS pokouší připojit
    request.IdUser = idUser;

    //zavolání vlastní WS
    var response = Client.ws_tryLogin(request);

    Correlate(request, response);

    //pokud je navracený chybový stav
    if (response.Acknowledge == AcknowledgeType.Failure)
    {
        //vyvolej vyjímku
        throw new ApplicationException(response.Message);
    }

    //vrať odpověď
    return response;
}

/// <summary>
/// Nactení dostupnych uzivatelu pro klienta WS
/// </summary>
/// <returns></returns>
public tUserDTO[] Get_Available_Users()
{
    //vytvoření requestu
    RequestBase request = new RequestBase().Prepare();

    //ziskani dostupnych uzivatelu
    var response = Client.ws_drazba_get_AvailableUsers(request);

    Correlate(request, response);

    //pokud je navracený chybový stav
    if (response.Acknowledge == AcknowledgeType.Failure)
    {
        //vyvolej vyjímku
        throw new ApplicationException(response.Message);
    }

    return response.tUsers;
}

/// <summary>
/// Nactení dostupnych vyhledavacich profilu pro uzivatele
/// </summary>
/// <param name="idUser"></param>
/// <returns></returns>
public tSearchProfileDTO[] Get_SearchProfiles_For_User(int idUser)
{
    //vytvoření requestu
```

```
tSearchProfileRequest request = new tSearchProfileRequest().Prepare();

//nastaveni uzivatele, pro ktereho pozadujeme vratit seznam vyhledavacich profilu
request.IdUser = idUser;

//zavolani WE
var response = Client.ws_drazba_get_SearchProfiles_For_User(request);

Correlate(request, response);

//pokud je navraceny chybovy stav
if (response.Acknowledge == AcknowledgeType.Failure)
{
    //vyvolej vyjimku
    throw new ApplicationException(response.Message);
}
return response.tSearchProfiles;
}

/// <summary>
/// Metoda pro vytvoreni requestu pro volani metody parovani z povinných atributů
/// </summary>
/// <param name="IdUser">Id uzivatele</param>
/// <param name="MaxAdvertAgeDays">maximalni stari inzeratu</param>
/// <param name="IdCategory"></param>
/// <param name="IdOfferType"></param>
/// <param name="IdCommercialMode"></param>
/// <param name="IdCountry"></param>
/// <returns></returns>
public ws_pairing_Request PreparePairingRequest(int IdUser,
                                               int MaxAdvertAgeDays,
                                               int IdCategory,
                                               int IdOfferType,
                                               int IdCommercialMode,
                                               int IdCountry)
{
    //vytvor request
    ws_pairing_Request request = Prepare(new ws_pairing_Request());

    request.IdUser = IdUser;
    request.ResultAgeDays = MaxAdvertAgeDays;
    request.CategoryId = new PairingAttributeCategoryIdDTO() { Category = new CategoryDTO() {
IdCategory = IdCategory } };
    request.OfferTypeId = new PairingAttributeOfferTypeIdDTO() { OfferType = new OfferTypeDTO()
{OfferType = IdOfferType } };
    request.CommercialModeId = new PairingAttributeCommercialModeIdDTO() { CommercialMode = new
CommercialModeDTO() { IdCommercialMode = IdCommercialMode } };
    request.CountryId = new PairingAttributeCountryIdDTO() { Country = new CountryDTO() { IdCountry =
IdCountry } };

    return request;
}

/// <summary>
/// Metoda vykonavajici vlastni parovani
/// </summary>
/// <param name="request"></param>
/// <returns></returns>
public tHitDTO[] ProcessPairing(ws_pairing_Request request)
{
    //zavolej WS
    var response = Client.ws_drazba_pairing(request);

    Correlate(request, response);

    //pokud je navraceny chybovy stav
    if (response.Acknowledge == AcknowledgeType.Failure)
    {
```

```
        //vyvolej vyjimku
        throw new ApplicationException(response.Message);
    }

    return response.tHits;
}

/// <summary>
/// Nacteni ciselniku sub-regionu
/// </summary>
/// <param name="idRegion"></param>
/// <returns></returns>
public SubRegionDTO[] Get_LOV_SubRegion(int? idRegion)
{
    //vytvoření requestu
    LOV_SubRegionRequest request = new LOV_SubRegionRequest().Prepare();

    //nastavení filtru podle regionu
    request.IdRegion = idRegion;

    var response = Client.ws_octop_get_LOV_SubRegion (request);

    Correlate(request, response);

    //pokud je navracený chybový stav
    if (response.Acknowledge == AcknowledgeType.Failure)
    {
        //vyvolej vyjimku
        throw new ApplicationException(response.Message);
    }

    return response.SubRegions;
}

/// <summary>
/// Nacteni ciselniku regionu
/// </summary>
/// <param name="idRegion"></param>
/// <returns></returns>
public RegionDTO[] Get_LOV_Region(int? idRegion)
{
    //vytvoření requestu
    LOV_RegionRequest request = Prepare(new LOV_RegionRequest());

    request.IdRegion = idRegion;

    var response = Client.ws_octop_get_LOV_Region (request);

    Correlate(request, response);

    //pokud je navracený chybový stav
    if (response.Acknowledge == AcknowledgeType.Failure)
    {
        //vyvolej vyjimku
        throw new ApplicationException(response.Message);
    }

    return response.Regions;
}
}
```


3 WEBOVÉ SLUŽBY PRO DRAŽBY

3.1 Seznam použitých webových služeb

Jméno služby	Poznámka
<code>ws_drazba_2c</code>	Služba pro získávání záznamů (inzerátů) monitoringu dražených nemovitostí a to od specifikovaného data
<code>ws_octop_get_AvailableUsers</code>	Služba pro načtení seznamu uživatelů, na které jsou přidělena přístupová práva
<code>ws_octop_get_LOV_Region</code>	Služba pro získání číselníku regionů (krajů). Vstupním parametrem může být identifikátor regionu - hodnoty číselníku, který slouží jako filtr
<code>ws_octop_get_LOV_SubRegion</code>	Služba pro získání číselníku sub-regionů (okresů). Vstupním parametrem může být identifikátor sub-regionu - hodnoty číselníku, který slouží jako filtr.

3.2 ws_drazba_2c

```
[WsdDocumentation("Služba pro získávání záznamů monitoringu Dražeb od specifikovaného data.")]
```

```
tHitResponse ws_drazba_2c(ws_drazba_2c_Request request);
```

Metoda poskytuje kolekci záznamů `tHit` pro specifikovaného uživatele a specifikované parametry výběru. Je vrácena pouze ta část existujících záznamů, které jsou mladší než datum reprezentovaný vstupní hodnotou `HitDateLast`, a které vyhovují parametrům výběru.

Pokud by počet navrácených rekordů převyšoval hodnotu **TopSize**, bude navrženo **TopSize** záznamů.

ws_drazba_2c_Request – vstupní parametry

Název elementu	Typ	Možné hodnoty	Poznámka
<code>IdUser</code>	int	-	Id uživatele, který požaduje dotaz (parametr vymezuje jeho kompetence pro obdržení dat)
<code>oSearchProfileRequest</code>	tSearchProfileDrazbaRequestDTO	-	kolekce parametrů specifikující výběrová kritéria. Podrobný popis je v kapitole tSearchProfileDrazbaRequestDTO

tHitDrazbaResponse – výstupní parametry

Název elementu	Typ	Možné hodnoty	Poznámka
<code>tHits</code>	tHitDrazbaDTO	-	kolekce tHitDrazbaDTO – seznam vyhledaných záznamů (Dražeb) odpovídajících výběrovým kritériím reprezentovaným vstupními parametry.

3.2.1 Ukázka volání funkce ws_drazba_2c

```
ws_drazba_2c_Request RQ = new ws_drazba_2c_Request();

//nastavení uživatelských údajů
RQ.IdUser = Convert.ToInt32(tbIDUser.Text);
RQ.ClientUsername = tbClientUsername.Text;
RQ.ClientTag = tbClientTag.Text;
RQ.ClientPassword = tbClientPassword.Text;

RQ.oSearchProfileRequest = new tSearchProfileDrazbaRequestDTO();

List<tSearchProfileCategoryRequestDTO> Cats = new
List<tSearchProfileCategoryRequestDTO>();
```

```
    ///Nastavení kategorií
    foreach(ListItem i in chbCategorie.Items)
    {
        if(i.Selected==true)
        {
            Cats.Add(new tSearchProfileCategoryRequestDTO() { IdCategory =
Convert.ToInt32(i.Value) });
        }
    }
    RQ.oSearchProfileRequest.SearchProfileCategoriesRequest = Cats.ToArray();

    ///nastavení krajů
    List<tSearchProfileRegionRequestDTO> Regions = new
List<tSearchProfileRegionRequestDTO>();
    foreach (ListItem i in chbKraje.Items)
    {
        if (i.Selected == true)
        {
            Regions.Add(new tSearchProfileRegionRequestDTO() { IdRegion =
Convert.ToInt32(i.Value) });
        }
    }
    ///nastavení okresů
    foreach (ListItem i in chbOkresy.Items)
    {
        if (i.Selected == true)
        {
            Regions.Add(new tSearchProfileRegionRequestDTO() { IdSubregion =
Convert.ToInt32(i.Value) });
        }
    }

    RQ.oSearchProfileRequest.SearchProfileRegionsRequest=Regions.ToArray();

    ///příznak, zda vyžadují pouze aktuální dražby
    RQ.oSearchProfileRequest.OnlyActual = chbOnlyActual.Checked;

    ///omezení na nejstarší dražbu
    if (!string.IsNullOrEmpty(tbHitDateLast.Text))
    {
        RQ.oSearchProfileRequest.HitDateLast =
Convert.ToDateTime(tbHitDateLast.Text);
    }
    ///omezení počtu vrácených výsledků
    if (!string.IsNullOrEmpty(tbTopSize.Text))
    {
        RQ.oSearchProfileRequest.TopSize = Convert.ToInt32(tbTopSize.Text);
    }

    OctopusServiceClient Client = new OctopusServiceClient();

    ///volání samotné WS
    tHitDrazbaResponse Response= Client.ws_drazba_2c(RQ);

    ///neúspěšné volání služby - vypsání chybového hlášení
    if (Response.Acknowledge == AcknowledgeType.Failure)
    {
        ltResult.Text = "Chyba + " + Response.Message;
    }
    ///Načtení a vykreslení vrácených dat
    if(Response.tHits!=null)
    {
        ltResult.Text = "Volání bylo úspěšné. Celkem vráceno " +
Response.tHits.Count();
        rpt_AdvertsWithPaging.DataSource = Response.tHits;
        rpt_AdvertsWithPaging.DataBind();
    }
    else
```

```
{
    ltResult.Text = "Nejsou k dispozici žádná data";
}
```

3.3 ws_octop_get_AvailableUsers

[WsdDocumentation("Služba pro načtení seznamu uživatelů, na které jsou přidělena přístupová práva")]

tUserResponse ws_octop_get_AvailableUsers(RequestBase request);

Metoda pro získání seznamu uživatelů, na které jsou přidělena práva. Služba nemá žádné speciální vstupní parametry.

tUserResponse – výstupní parametry

Název elementu	Typ	Možné hodnoty	Poznámka
tUsers	tUserDTO[]	-	kolekce tUserDTO – seznam uživatelů – viz datové typy

3.4 ws_octop_get_LOV_Region

[WsdDocumentation("Načtení číselníků - Region")]

LOV_RegionResponse ws_octop_get_LOV_Region(LOV_RegionRequest request);

Služba pro získání číselníku regionů. Vstupním parametrem může být identifikátor regionu - hodnoty číselníku, který slouží jako filtr.

LOV_RegionRequest – vstupní parametry

Název elementu	Typ	Možné hodnoty	Poznámka
IdRegion	Int?	-	Identifikátor regionu

LOV_RegionResponse – výstupní parametry

Název elementu	Typ	Možné hodnoty	Poznámka
Regions	RegionDTO[]	-	Pole RegionDTO – seznam hodnot číselníku regionů

3.5 ws_octop_get_LV_SubRegion

[WsdDocumentation("Načtení číselníků - SubRegion")]

LOV_SubRegionReponse ws_octop_get_LOV_SubRegion(LOV_SubRegionRequest request);

Služba pro získání číselníku subregionů. Vstupním parametrem může být identifikátor subregionu - hodnoty číselníku, který slouží jako filtr.

LOV_SubRegionRequest – vstupní parametry

Název elementu	Typ	Možné hodnoty	Poznámka
IdSubRegion	Int?	-	Identifikátor sub regionu

LOV_SubRegionReponse – výstupní parametry

Název elementu	Typ	Možné hodnoty	Poznámka
SubRegions	SubRegionDTO[]	-	Pole SubRegionDTO – seznam hodnot číselníku regionů

4 DATOVÉ TYPY - DATA TRANSFER OBJEKTY

4.1 tHitDrazbaDTO

Transfer objekt pro záznam monitoringu realit.

Název elementu	Typ	Možné hodnoty	Poznámka
IdHit	int	-	PK tabulky - Id záznamu Dražeb
Type	int	3	Typ záznamu – Dražby
IdCategory	int	CC_Category	Kategorie nemovitosti
SubCategory	int	-	nevyužito
OfferType	int	1 nebo NULL	Nevyužito
CommercialMode	int	1 nebo NULL	nevyužito
IdRegion	Int?	CC_Region	Specifikuje kod kraje, kde leží dražená nemovitost
SubRegion	Int?	CC_SubRegion	Specifikuje kod okresu, kde leží dražená nemovitost
IdPortal	int	Číselník	Id zdroje, kde byla publikována informace o dražené nemovitosti
BodyText	string		Text o dražené nemovitosti Pokud neexistuje text, je zde číslo dražební vyhlášky.
BodyStampHash	int	0	nevyužito
Phone1	Long?	NULL	nevyužito
Phone2	Long?	NULL	nevyužito
Phone3	Long?	NULL	nevyužito
DateOfFirstRelease	DateTime?		Datum, kdy se bude konat dražba
DateOfLastChange	DateTime?	NULL	nevyužito
DetailURL	string		Url k detailu Dražené nemovitosti
ClientSMS	string		Text s hlavními informacemi k dražené nemovitosti - CAPTION
Price	int?		Navržená dražební cena
DateOfGrab	Datetime		Datum posledního vygravování Dražby
AdvertID	string	-	Číslo dražby, dražební vyhlášky
IdRA	int?	NULL	nevyužito
Locality	string		Textový popis lokality, kde se dražební nemovitost nachází
Area	int?	NULL	Nevyužito
nameRA	string		Link na dražební vyhlášku – obvykle PDF soubor
IdCountry	int?	CC_Country	Vždy hodnota 1
IdVlastnictvi	int?	NULL	Nevyužito
HitRepeatCount	Int?	NULL	Nevyužito
Email	string	NULL	Nevyužito
Publisher	string	NULL	Nevyužito
PhoneRepeatCount	int?	NULL	Nevyužito
jtskx	Long?	NULL	Nevyužito
jtsky	Long?	NULL	Nevyužito
kodKatastralUzemi	Int?	NULL	Nevyužito
Obec	string	NULL	Název obce, kde se nachází dražená nemovitost
IsProcessed	Bit?	NULL	Nevyužito

4.2 tSearchProfileDrazbaRequestDTO

Request objekt definující vyhledávací parametry pro výběr kolekce záznamů.

Název elementu	Typ	Poznámka
----------------	-----	----------

OnlyActual	Bool?	V případě True , vrátí pouze aktuální Dražby
SearchProfileCategoriesRequest	tSearchProfileCategoryRequestDTO[]	Zvolené kategorie nemovitostí, na kterých je Dražba – pokud bude pole prázdné, budou vráceny dražby ze všech kategorií
SearchProfileRegionsRequest	tSearchProfileRegionRequestDTO[]	Zvolené regiony (kraje + okresy), kde je sledována Dražba nemovitostí. Pokud bude pole prázdné, budou vráceny dražby ze všech regionů (sub-regionů)
HitDateLast	DateTime	Položka HitDateLast reprezentuje hodnotu tHitDTO.DateOfGrab - datum pořízení záznamu o dražbě. Výsledná kolekce navrátí pouze takové tHit, které jsou mladší než HitDateLast.
TopSize	Int?	Maximální počet vrácených záznamů –default 1000

4.3 tSearchProfileCategoryRequestDTO

Dotaz objekt pro specifikaci typu nemovitosti a typu inzerátu.

Název elementu	Typ	Možné hodnoty	Poznámka
IdCategory	Int	CC_Category	Id kategorie nemovitosti

4.4 tSearchProfileRegionRequestDTO

Dotaz objekt pro specifikaci kraje a okresu nemovitosti

Název elementu	Typ	Možné hodnoty	Poznámka
IdRegion	Int?	CC_Region	Id regionu (kraje)
IdSubRegion	Int?	CC_SubRegion	Id sub regionu (okresu)

4.5 tUserDTO

Transfer objekt pro uživatele.

Název elementu	Typ	Možné hodnoty	Poznámka
IdUser	int	PK tabulky	Id uživatele
IdRealAgencyClient	int	PK tabuky	Id společnosti (pobočky)
Name	string	-	Jméno uživatele
Email	string	-	Email
Phone	Int?	-	Tel. číslo
UserName	string	-	Uživatelské jméno
IsActive	bool	-	Je uživatel aktivní?
Role	int	CC_Role	Role uživatele

5 ČÍSELNÍKY.

5.1 CC_Category

IdCategory	Category	CategoryName	CategoryAbbrev	CategoryDescription
1	1	Byty	BYT	
2	2	Domy	DUM	Rodinné domy, činžovní domy, hotely
4	4	Pozemky	POZ	Stavební pozemky, pole, lesy
5	8	Chaty	CHAT	
6	16	Komerční prostory	KP	
7	32	Kanceláře	KANC	
8	64	Garáže	GAR	
9	128	Developerské projekty	DEV	

5.2 CC_Region

IdRegion	Region	RegionName
1	1	Praha
2	2	Středočeský
3	4	Jihočeský
4	8	Plzeňský
5	16	Karlovarský
6	32	Ústecký
7	64	Liberecký
8	128	Královéhradecký
9	256	Pardubický
10	512	Vysočina
11	1024	Jihomoravský
12	2048	Olomoucký
13	4096	Moravskoslezský
14	8192	Zlínský

5.2.1 Abecední řazení

IdRegion	Region	RegionName
3	4	Jihočeský
11	1024	Jihomoravský
5	16	Karlovarský
8	128	Královéhradecký
7	64	Liberecký
13	4096	Moravskoslezský
12	2048	Olomoucký
9	256	Pardubický
4	8	Plzeňský
1	1	Praha
2	2	Středočeský
6	32	Ústecký
10	512	Vysočina
14	8192	Zlínský

5.3 CC_SubRegion

IdSubRegion	IdRegion	RegionName	IdSubRegion	IdRegion	RegionName
105	1	nepřiřazeno	30	13	Nový Jičín
49	1	Praha 1	31	13	Opava
58	1	Praha 10	89	13	Ostrava
50	1	Praha 2	32	13	Ostrava - město
51	1	Praha 3	84	14	Kroměříž
52	1	Praha 4	118	14	nepřiřazeno
53	1	Praha 5	85	14	Uherské Hradiště
54	1	Praha 6	86	14	Vsetín
55	1	Praha 7	87	14	Zlín
56	1	Praha 8	208	15	Bratislava
57	1	Praha 9	197	15	Bratislava I
88	1	Praha hl.m.	198	15	Bratislava II
59	2	Benešov	199	15	Bratislava III
60	2	Beroun	200	15	Bratislava IV
61	2	Kladno	201	15	Bratislava V
62	2	Kolín	119	15	Malacky
63	2	Kutná Hora	122	15	nepřiřazeno
64	2	Mělník	120	15	Pezinok
65	2	Mladá Boleslav	121	15	Senec
106	2	nepřiřazeno	124	16	Dunajská Streda
66	2	Nymburk	125	16	Galanta
69	2	Praha-východ	126	16	Hlohovec
70	2	Praha-západ	130	16	nepřiřazeno
68	2	Příbram	127	16	Piešťany
71	2	Rakovník	128	16	Senica

1	3	České Budějovice		129	16	Skalica
2	3	Český Krumlov		123	16	Trnava
3	3	Jindřichův Hradec		132	17	Bánovce nad Bebravou
107	3	nepřirazeno		133	17	Ilava
4	3	Písek		196	17	Myjava
5	3	Prachatice		139	17	nepřirazeno
6	3	Strakonice		134	17	Nové Mesto nad Váhom
7	3	Tábor		135	17	Partizánske
42	4	Domažlice		137	17	Považská Bystrica
43	4	Klatovy		136	17	Prievidza
108	4	nepřirazeno		138	17	Púchov
90	4	Plzeň		131	17	Trenčín
46	4	Plzeň-jih		141	18	Komárno
44	4	Plzeň-město		142	18	Levice
45	4	Plzeň-sever		147	18	nepřirazeno
47	4	Rokycany		140	18	Nitra
48	4	Tachov		143	18	Nové Zámky
15	5	Cheb		144	18	Šaľa
16	5	Karlovy Vary		145	18	Topoľčany
109	5	nepřirazeno		146	18	Zlaté Moravce
17	5	Sokolov		149	19	Bytča
72	6	Děčín		150	19	Čadca
73	6	Chomutov		151	19	Dolný Kubín
74	6	Litoměřice		152	19	Kysucké Nové Mesto
75	6	Louny		153	19	Liptovský Mikuláš
76	6	Most		154	19	Martin
110	6	nepřirazeno		155	19	Námestovo
77	6	Teplice		159	19	nepřirazeno
78	6	Ústí nad Labem		156	19	Ružomberok
23	7	Česká Lípa		157	19	Turčianske Teplice
93	7	Jablonec n/N		158	19	Tvrdošín
24	7	Jablonec nad Nisou		148	19	Žilina
25	7	Liberec		160	20	Banská Bystrica
111	7	nepřirazeno		161	20	Banská Štiavnica
26	7	Semily		162	20	Brezno
18	8	Hradec Králové		163	20	Detva
19	8	Jičín		164	20	Krupina
20	8	Náchod		165	20	Lučenec
112	8	nepřirazeno		173	20	nepřirazeno
21	8	Rychnov nad Kněžnou		166	20	Poltár
22	8	Trutnov		167	20	Revúca
38	9	Chrudim		168	20	Rimavská Sobota
113	9	nepřirazeno		169	20	Veľký Krtíš
39	9	Pardubice		170	20	Zvolen
40	9	Svitavy		171	20	Žarnovica
41	9	Ústí nad Orlicí		172	20	Žiar nad Hronom
79	10	Havlíčkův Brod		175	21	Bardejov
80	10	Jihlava		176	21	Humenné
114	10	nepřirazeno		177	21	Kežmarok
81	10	Pelhřimov		178	21	Levoča
82	10	Třebíč		179	21	Medzilaborce
83	10	Žďár nad Sázavou		187	21	nepřirazeno
8	11	Blansko		180	21	Poprad
92	11	Brno		174	21	Prešov
10	11	Brno-město		181	21	Sabinov
9	11	Brno-venkov		182	21	Snina
11	11	Břeclav		183	21	Stará Ľubovňa
12	11	Hodonín		184	21	Stropkov
115	11	nepřirazeno		185	21	Svidník
13	11	Vyškov		186	21	Vranov nad Topľou

14	11	Znojmo		188	22	Gelnica
33	12	Jeseník		202	22	Košice I
116	12	nepřiřazeno		203	22	Košice II
34	12	Olomouc		204	22	Košice III
35	12	Prostějov		205	22	Košice IV
36	12	Přerov		206	22	Košice V
37	12	Šumperk		207	22	Košice-okolie
27	13	Bruntál		189	22	Michalovce
28	13	Frýdek Místek		190	22	Moldava nad Bodvou
29	13	Karviná		195	22	nepřiřazeno
117	13	nepřiřazeno		191	22	Rožňava
				192	22	Sobrance
				193	22	Spišská Nová Ves
				194	22	Třebíšov

6 UKÁZKA ČISTÉHO XML DOTAZU NA WEBOVOU SLUŽBU WS_DRAZBA_2C

```

<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <ws_drazba_2c xmlns="http://octopuspro.cz/OctopusProService">
      <Request xmlns:a="http://www.octopuspro.cz/types/"
xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
        <a:Action i:nil="true"/>
        <a:ClientPassword>xxxx
        </a:ClientPassword>
        <a:ClientTag>xxxxx
        </a:ClientTag>
        <a:ClientUsername>xxxxx
        </a:ClientUsername>
        <a:LoadOptions i:nil="true"
xmlns:b="http://schemas.microsoft.com/2003/10/Serialization/Arrays"/>
        <a:RequestId i:nil="true"/>
        <a:IdUser>xxxxx
        </a:IdUser>
        <a:oSearchProfileRequest
xmlns:b="http://schemas.datacontract.org/2004/07/Octop.ActionService.Message.Drazby">
          <b:HitDateLast>0001-01-01T00:00:00
          </b:HitDateLast>
          <b:OnlyActual>>false
          </b:OnlyActual>
          <b:SearchProfileCategoriesRequest
xmlns:c="http://schemas.datacontract.org/2004/07/Octop.ActionService.Message">
            <c:tSearchProfileCategoryRequestDTO>
              <c:IdCategory>1
              </c:IdCategory>
            </c:tSearchProfileCategoryRequestDTO>
          </b:SearchProfileCategoriesRequest>
          <b:SearchProfileRegionsRequest
xmlns:c="http://schemas.datacontract.org/2004/07/Octop.ActionService.Message">
            <c:tSearchProfileRegionRequestDTO>
              <c:IdRegion>1
              </c:IdRegion>
              <c:IdSubregion i:nil="true"/>
            </c:tSearchProfileRegionRequestDTO>
          </b:SearchProfileRegionsRequest>
        </a:oSearchProfileRequest>
      </Request>
    </ws_drazba_2c>
  </s:Body>
</s:Envelope>

```



```
    <b:TopSize>100
  </b:TopSize>
</a:oSearchProfileRequest>
</Request>
</ws_drazba_2c>
</s:Body>
</s:Envelope>
```